END

FILMED

DTIC

MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU OF STANDARDS-1963-A

BR93478 ①

AD–A148 489

RSRE
MEMORANDUM No. 3739

ROYAL SIGNALS & RADAR ESTABLISHMENT

A FORTRAN PLOTTING PACKAGE FOR GRAPHIC VDUs

Authors: V J Mifsud and
C Broughton

PROCUREMENT EXECUTIVE,
MINISTRY OF DEFENCE,
RSRE MALVERN,
WORCS.

DTIC FILE COPY

DTIC
ELECTE
DEC 1 2 1984
S D
E

ROYAL SIGNALS AND RADAR ESTABLISHMENT

Memorandum   3739

Title: A FORTRAN PLOTTING PACKAGE FOR GRAPHIC VDUs

Authors: V J Mifsud, C Broughton

Date: August 1984

Summary

A set of plotting routines have been implemented in FOR-
TRAN on DEC LSI-11, PDP-11 and VAX-11 computers to provide a
transportable graphics capability to improve data presentation
and machine control on vector scan particle beam lithography ma-
chines. The routines are general in nature and have already
proved to be of use in many scientific applications requiring
compact and tailored graphics capabilities within specific pro-
grams. This document is designed to act as a User's Guide for
the suite. Versions of the routines are available for ReGIS,
Tektronics 4010/4014 and SIGMA native-mode graphics terminals.

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	

A FORTRAN PLOTTING PACKAGE
FOR GRAPHIC VDUs
USERS GUIDE
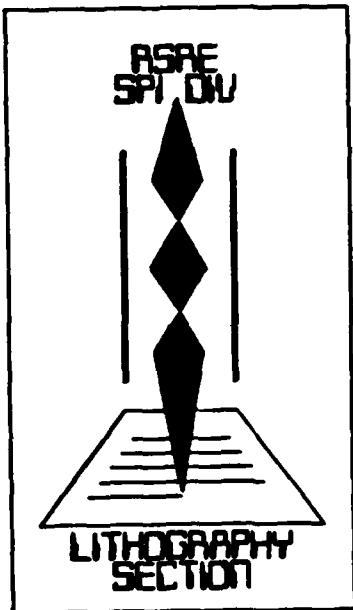
V J Mifsud, C Broughton
Lithography Section
SP1 Division
RSRE Malvern

UNLIMITED

1. Introduction

The specific requirements of pattern verific-
ation and wafer layout control in lithographic
applications requires a limited set of graphics routi-
nes to improve data presentation and to ease machine
operation. The specific requirements have been defined
and a suite of subroutines written to satisfy this
need. The suite is highly modular and is written in a
high-level language to ease its adaptation to suit a
variety of tasks.

This package is intended to be used in conjunc-
tion with both monochrome and colour terminals. The
former include the DEC VT125, VT24X and Tektronics
4014-compatible graphics terminals (including the
Pericom series, the VT24X and VT100 type terminals up-
graded with the Selanar graphics option), the latter
include the SIGMA range of compatible terminals and
the VT241. It is coded mainly in FORTRAN-77 with
MACRO-11 used where required for speed. The suite of
subroutines is designed to run on RSX-11M systems on
PDP-11 or LSI-11 computers. A VAX VMS version is also
available. Conversion to run on other terminals or op-
erating systems should be easily achievable due to the
use of high level routines and a modular structure.

2. Overview

The package is designed to present a set of subrouti-
nes that include a one to one correspondence with the
DEC supplied routines for the Servogor 281 plotter
(also marketed by Philips and Calcomp). The intention
is to permit programs requiring graphic output to have
direct access to the Servogor plotter, a ReGIS termi-
nal or a Tektronics PLOT-10 compatible terminal.
Identical calls are provided as appropriate, requiring
only that a graphics program be task built with the
appropriate libraries to suit the specific terminal
type. In addition, further routines provide a broader
range of functions, though compatibility with the
plotter is then not available.

The package expects its coordinates (either:
user's or the terminal's) in 0.1 mm units (integers)
and does not accepts real numbers denoting cm. Two
types of routines are provided, those that correspond
to an A4 page (30 by 21 cm), and those that use the
coordinates of the terminal directly.

The package can be used in FORTRAN-77 under IAS ,
all RSX11M/M+ systems, RT11 and VMS. It is possible to
modify the suite to run under FORTRAN-IV and IV PLUS
if required. This description refers specifically to
RSX-11M operating systems.

Implementation of these FORTRAN routines in an
operational environment is a prerequisite for usage of
the any of the higher software level (written normally
in FORTRAN), e.g.: the HCFS-Package (Hardware Com-
patible Fortran Software) and 2-D and 3-D plotting
packages already available. Mixed usage of subrouti-
nes from different software levels, in the same
application program is also possible but should be
checked thoroughly.

2.1. Index of subroutines

The package subroutines can logically be
divided into functional groups, as follows.

2.1.1. GROUP 1 : Vectors

```
RMOVE,JMOVE       relative move,
     A4 coords option
XRMOVE,XJMOVE     relative move,
     graphic terminal coords option
RPLOT,JPLOT       relative plot,
     A4 coords option
XRPLOT,XJPLOT     relative plot,
     graphic terminal coords option
AMOVE,KMOVE       absolute move,
```

```
                    A4 coords option
        XAMOVE,XKMOVE    absolute move,
            graphic terminal coords option
        APLOT,KPLOT      absolute plot,
            A4 coords option
        APLOT,KPLOT      absolute plot,
            graphic terminal coords option
```

2.1.2. GROUP 2 : Pen and line type

    PENUP   dummy in monochrome terminals,
for compatibility
    NEWPEN dummy in monochrome terminals,
change colour in colour terminals
    PLUMA   dummy in monochrome terminals,
change colour in colour terminals
    LINTYP select line type

2.1.3. GROUP 3 : Text plotting

```
        SETCHR   select character size and angle
        NEWCHR   select character set
        SETSLN   select character slant
        POINT    plot a point mark
        TEXT     plot text
```

2.1.4. GROUP 4 : Circle, sector and axes

```
        CIRC     plot circle
        SECT     plot sector
        AXEL     plot axis
```

2.1.5. GROUP 5 : Digitising and window

    LOCAT   get coordinates without user
intervention
    DIGTZE   digitize interactively (not
available on VT125)
    OFFSET select offset coordinates
    WINDOW select plotting window
    PLTWND   verify the current plotting
window

2.1.6. GROUP 6 : Miscellaneous

```
        CHART    dummy, for compatibility
        PLTON    open output channel  and  ini-
```

tialize graphic terminal
PLTOFF   close O/P channel and logically
disconnect
OFFBUF   dummy, graphic terminal used in
real time
ONBUF   dummy, graphic terminal used in
real time
PLTNAM   change default names (O/P
device and file)
PLTERR transfer ERROR condittion to
user program

2.1.7.

GROUP 7 : Extensions

ARECT   Rectangle drawing routine, A4
coordinates
XARECT   Rectangle drawing routine,
plotter coords
FILSEL Sets to filled or edged rectan-
gle drawing

CURSOR Returns cursor coordinates and
key code

ERALN Erases a previously drawn line
(A4 coords)
XERALN Erases a previously drawn line
(Terminal units)
ERART Erases a previously drawn rec-
tangle (A4 coords)
XERART Erases a previously drawn rec-
tangle (Term units)
ERASE Selects normal or erase writing
for subsequent features

2.2. Notes on usage

This package, with the exception of the
routines LOCAT, CURSOR, and DIGTZE, can also be
used with the graphic terminal offline instead
of on-line, since all the other routines are
output only. This makes it possible to   store
pictures (plots) in files for later plotting.
In this memorandum, online plotting is defined
as plotting with the graphic terminal directly
connected to a dedicated line, with the plot-
ting program outputting the graphic commands to

the terminal which immediately begins executing
them. Offline plotting is considered to be
plotting into a file, which later on is output
to the graphic terminal by using the appro-
priate file transfer utility program: PLTPER or
PLTVT.

3. Detailed description of the package subroutines

These subroutines will be described complete with variable usage, call and ,if possible, examples. The order will be the same as in 2.1.

Equivalent subroutine names are mentioned in parenthesis.

RMOVE (JMOVE)

This subroutine moves the cursor with relative coordinates, within the limits of an A4 page (30cm X 21cm or 3000 X 2100 graphic units).

Use : CALL RMOVE (IX,IY)

    IX   = x-coordinate
    IY   = y-coordinate

XRMOVE (XJMOVE)

This subroutine moves the cursor with relative coordinates, within the limits of the particular graphic terminal.

    For a VT125/24X, this is 680 X 480 units.
    For a Pericom or Tektronics equivalent,
            this is 1024 X 780 units.
    For a SIGMA colour terminal,
            this is 768 X 512 units.

Use : CALL XRMOVE (IX,IY)

    IX   = x-coordinate
    IY   = y-coordinate

RPLOT (JPLOT)

This subroutine plots a line with relative coordinates, within the limits of an A4 page (30cm X 21cm or 3000 X 2100 graphic units).

Use : CALL RPLOT (IX,IY)

    IX   = x-coordinate
    IY   = y-coordinate

XRPLOT (XJPLOT)

This subroutine plots a line with relative coordinates, within the limits of the particular graphic terminal.

      For a VT125/24X, this is 680 X 480 units.
      For a Pericom or Tektronics equivalent,
           this is 1024 X 780 units.
      For a SIGMA colour terminal,
           this is 768 X 512 units.

Use : CALL XRPLOT (IX,IY)

     IX   = x-coordinate
     IY   = y-coordinate

AMOVE (KMOVE)

This subroutine moves the pen with absolute coordinates, within the limits of an A4 page (30cm X 21cm or 3000 X 2100 graphic units).

Use : CALL AMOVE(IX,IY)

     IX   = x-coordinate
     IY   = y-coordinate

XAMOVE (XKMOVE)

This subroutine moves the pen with absolute coordinates, within the limits of the particular graphic terminal.

      For a VT125/24X, this is 680 X 480 units.
      For a Pericom or Tektronics equivalent,
           this is 1024 X 780 units.
      For a SIGMA colour terminal,
           this is 768 X 512 units.

Use : CALL XAMOVE(IX,IY)

     IX   = x-coordinate
     IY   = y-coordinate

APLOT (KPLOT)

This subroutine plots a line with absolute coordinates, within the limits of an A4 page (30cm X 21cm or 3000 X 2100 graphic units).

Use : CALL APLOT(IX,IY)

    IX   = x-coordinate
    IY   = y-coordinate

XAPLOT (XKPLOT)

This subroutine plots a line with absolute coordinates, within the limits of the particular graphic terminal.
    For a VT125/24X, this is 680 X 480 units.
    For a Pericom or Tektronics equivalent,
            this is 1024 X 780 units.
    For a SIGMA colour terminal,
            this is 768 X 512 units.

Use : CALL XAPLOT(IX,IY)

    IX   = x-coordinate
    IY   = y-coordinate

PENUP

This subroutine is a dummy, to match the plotter routine that raises or lowers the pen

Use : CALL PENUP(IM)

    IM   = 0 : pen up
    IM   = 1 : pen down

NEWPEN

In the plotter suite, this subroutine selects one of the 8 pens on the plotter. It is also possible to park the current pen without selecting a new pen.
In this suite, it is a dummy routine for monochrome terminals. For colour terminals, the subroutine provides colour selection. The colour selection depends on the type of terminal used.

Use : CALL NEWPEN(IP)

    IP   = 0   : parks current pen (plotter only)
             1..8 : select pen, return to location
                  : or select colour, remain at loc-
ation

PLUMA

In the plotter package, this subroutine selects one of
the 8 pens on the plotter. It is also possible to
park the current pen without selecting a new
one.
In this suite, it is a dummy routine for monochrome
terminals. For colour terminals, the subroutine pro-
vides colour selection. The colour selected depends on
the type of terminal used.

Use : CALL PLUMA(IP)

    IP   = -1   : parks current pen (plotter
only)
               0...7 : select pen, return to location
                     : or select colour, remain at
location

LINTYP

This subroutine selects the line type and length of
elements if dashed.

Use : CALL LINTYP(IN,IL)

        IN   = 0 : continous line
                 1 : dotted line
                 2 : dashed line
                 3 : dashed dashed
                 4 : dashed dotted
        IL   = dummy, included for compatibility

SETCHR

This subroutine is used to set text and plot mark
characteristics, the detail of which depends on the

graphic terminal.

Use : CALL SETCHR(IH,ID,IW)

On a VT125,
IH   = character height
ID   = character direction in degrees
IW   = character width

On a Pericom,
IH   = character height
ID   = character direction in degrees
               (not implemented)
IW   = character width (not used)

On a SIGMA,
IH   = character height
ID   = character direction in degrees
               (0,90,180,270 available)
IW   = character width (not used)

NEWCHR

This subroutine selects one of five character sets of
the graphic terminal. The detail depends on the actual
terminal type.

Use : CALL NEWCHR(IN)

IN   = 0 : standard ASCII set
           1 : German set
           2 : Spanish set
           3 : Swedish-Finnish set
           4 : Danish-Norwegian set

SIGMA terminals do not offer this facility.

SETSLN

This subroutine is used to set the character slant.

Use : CALL SETSLN(IN)

IN   = 0 : 90 degrees (straight)
           1 : 75 degrees (forward slant)
        -90<IN<90 slant angle (forward or back)

POINT

This subroutine is used to plot a point mark

Use : CALL POINT(IN)

```
        IN      = 0...4 : plot point mark
                          (see figure 1 for marks)
```

TEXT

This subroutine plots text with a given number of characters.

Use : CALL TEXT(STRING ,IN)

```
        STRING  = string to be plotted (BYTE array)
        IN      = number of characters in STRING.
```

N.B. This routine requires the number of characters to be explicitly stated. The option to leave out the number of characters is not presently available.

CIRC

This subroutine plots a circle.

Use : CALL CIRC(IR)

```
        IR      = radius of circle.
                  Will plot the circle from the
                  perimeter point corresponding
                to 0 degrees
                  if > 0 : counterclockwise plot
                  if < 0 : clockwise plot
```

SECT

This subroutine will plot sectors of a circle.

Use : CALL SECT(IR,IA,IB)

        IR   = radius of circle of which the
                    sector is a part
                    if > 0 : counterclockwise plot
                    if < 0 : clockwise plot
        IA   = start angle (degrees)
        IB   = end angle (degrees)

AXEL

This subroutine plots a x- or y-axis with tic-marks.

Use : CALL AXEL(IM,IL,ID,IT1,IT2)

        IM   = select axis :
                    if 0 : x-axis
                    if 1 : y-axis
        IL   = absolute length of axis
        ID   = distance between tic-marks
                    if > 0 : right
                    if < 0 : left
        IT1   = length of first tic-mark
                    if > 0 : up
                    if < 0 : down
        IT2   = length of second tic-mark
                    if > 0 : up
                    if < 0 : down

    (See Figure 2 for illustration)

LOCAT

This subroutine is used to get the user's plotting
coordinates from the plotter. Note that it can only
be used with online plotting.

Use : CALL LOCAT(IX,IY)

   IX = integer variable to receive the x-coordinate
   IY = integer variable to receive the y-coordinate

DIGTZE

Subroutine to digitize interactively with the graphic
terminal, if this supports the feature (VT125 type
terminals do not). To use on the Pericom, either de-
press the optional light pen at the desired location,
or use the cursor positioning arrows to set the cursor
hairs onto the desired location and then depress the
space bar. This can only be used for online work. On
SIGMA terminals, the optional joystick is supported.

Use : CALL DIGTZE(IX,IY)

  IX = integer variable to receive the x-coordinate
  IY = integer variable to receive the y-coordinate

OFFSET

This subroutine is used to set a fixed x and/or y
offset for all subsequent graphical display.

Use : CALL OFFSET(IX,IY)

        IX   = x offset
        IY   = y offset

WINDOW

Subroutine to set the display window. All subsequent
plotting will only be recognized and plotted inside
this window. Note that this does not scale the coor-
dinates in any way.

Use : CALL WINDOW(IXMIN,IXMAX,IYMIN,IYMAX)

        IXMIN   = x-coordinate of lower left corner
        IXMAX   = x-coordinate of upper right corner
        IYMIN   = y-coordinate of lower left corner
        IYMAX   = y-coordinate of upper right corner

PLTWND

Subroutine to display the current window.

Use : CALL PLTWND

  no arguments

CHART

This routine is a null operation in the graphic terminals.

use : CALL CHART(ICM)

  ICM = advance paper ICM cm. Range : 1...64

PLTON

This subroutine opens the output channel (disc file or terminal line), verifies the output buffer and sends a string initialising the plotter.

Only subroutines PLTNAM and PLTERR can be called before PLTON is called.

Use : CALL PLTON (IBUF,LBUF [,LUN])

  IBUF = name of array reserved in user program
       and used as output buffer
  LBUF = length of array IBUF in words
      in range 42...512, if IBUF has length
      >512, space over 512 words is unused;
      optimal LBUF for disc access = 512
      words
  LUN = optional channel number used for ouput
      channel; default LUN=1 (hint: most
      users may prefer automatic channel
      allocation by system subroutine
      GTCHN to get a free channel number)

PLTOFF

This subroutine switches the graphic processor to logically off. The terminal will exit graphics mode after this call.

Use : CALL PLTOFF

        no arguments

OFFBUF

This subroutine switches buffering OFF, that means, every call to another subroutine causes immediate output of a string to the plotter. Initial software mode is BUFFERED. This call has no effect when writing to disc file.

use :   CALL OFFBUF

        no arguments

ONBUF

This subroutine switches buffering ON, that means, every call to another subroutine causes buffered output of a string to the plotter.

use :   CALL ONBUF

        no arguments

PLTNAM

This subroutine changes default names for output file and device and can be called only once and only before CALL START.

use :   CALL PLTNAM (FNAM,LFNAM,DNAM,LDNAM)

    FNAM = Array containing ASCII string
               of new file name
    LFNAM = length of new file name string
    DNAM = array containing ASCII string of new device
                   name
    LDNAM = length of device name

PLTERR

This subroutine establishes decoding of possible errors, when using this software package. Once called, it allows one to determine the nature of the fault. (Not fully implemented as not needed in this case).

Use :   CALL PLTERR (IER)

    IER = array of two words reserved for error
           information; lower word contains
           error number from plotter control
           processor; high word contains
            status information from file control
           processor; in both cases value equal
           zero mean success.

     Possible error number are:
         0 = no errors detected

ARECT

This routine draws the border or fills a rectangle, given the bottom left and top right coordinates in A4 units.

Use : CALL ARECT(IX1,IY1,IX2,IY2)

      IX1 = x coordinate of lower left corner
      IY1 = y coordinate of lower left corner
      IX2 = x coordinate of top right corner
      IY2 = y coordinate of top right corner

XARECT

This routine draws the border or fills a rectangle, given the bottom left and top right coordinates in terminal units.

Use : CALL XARECT(IX1,IY1,IX2,IY2)

      IX1 = x coordinate of lower left corner
      IY1 = y coordinate of lower left corner
      IX2 = x coordinate of top right corner
      IY2 = y coordinate of top right corner

FILSEL

This subroutine selects the drawing mode for rectangle
plotting. If set, the rectangles are filled, if clear-
ed, only the edges are drawn.

Use : CALL FILSEL(ILOG)

```
    ILOG  = BYTE variable. If set, block drawing
            is selected until changed.
            If cleared, edge drawing is
            selected until changed.
```

CURSOR

Operates as for DIGTZE, but it also returns the ASCII
code for the key depressed to transmit the current
cursor coordinates, as well as the coordinates them-
selves.

Use : CALL CURSOR(IX,IY,CHAR)

```
    IX    = X cursor coordinate
    IY    = Y cursor coordinate
    CHAR  = BYTE variable, containing the
            ASCII code of the key
```

ERALN

This routine allows the erasure of a previously drawn
line, that has been defined using A4 units. It is ne-
cessary to select the same line characteristics (ie
solid, dotted etc) as those used to draw the line
before calling this routine.

Use : CALL ERALN (IX,IY)

```
    IX  = X-coordinate (absolute)
    IY  = Y-coordinate (absolute)
```

A complement or erase line is drawn from the
current position to that defined by the subroutine ar-
guments.

XERALN

This routine allows the erasure of a previously drawn
line that has been defined using graphic terminal
units. It is necessary to select the same line charac-
teristics (ie solid, dotted etc) as those used to draw
the line before calling this routine.

Use : CALL XERALN (IX,IY)

        IX = X-coordinate (absolute)
        IY = Y-coordinate (absolute)

        A complement or erase line is drawn from the
current position to that defined by the subroutine ar-
guments.

ERART

This subroutine provides a means of erasing a
previously drawn rectangle, be it in border or filled-
in state. This routine accepts A4 coordinates, and is
complemented by the following routine that uses termi-
nal coordinates.

Use : CALL ERART (IX1,IY1,IX2,IY2)

  IX1 = X-coordinate of lower left corner (absolute)
  IY1 = Y-coordinate of lower left corner (absolute)
  IX2 = X-coordinate of top right corner (absolute)
  IY2 = Y-coordinate of top right corner (absolute)

        This routine uses complement drawing to erase
the figure. Consequently, it is necessary to choose
the same line or fill characteristics as were used to
draw the figure before calling this routine.

XERART

This subroutine provides a means of erasing a pre-
viously drawn rectangle, be it in border or filled-in
state. This routine accepts terminal coordinates, and
is complemented by the previous routine that uses A4
coordinates.

Use : CALL XERART (IX1,IY1,IX2,IY2)

IX1 = X-coordinate of lower left corner (absolute)
IY1 = Y-coordinate of lower left corner (absolute)
IX2 = X-coordinate of top right corner (absolute)
IY2 = Y-coordinate of top right corner (absolute)

This routine uses complement drawing to erase the figure. Consequently, it is necessary to choose the same line or fill characteristics as were used to draw the figure before calling this routine.

ERASE

This subroutine switches the terminal from normal to complement writing (and vice versa). It can be used to erase a block of lines or rectangles before returning to normal writing.

Use : CALL ERASE (ILOG)

ILOG = BYTE variable. If true, erase writing
is selected until ERASE is called
with ILOG set to false.
If false, normal writing is resumed.

4. An example using the package calls

```
C***   This is a program  in  FORTRAN-77  to  test  the
package calls
C
C*   Allocate 512 words buffer for output
C
      INTEGER IBUF(512)
C
C*   Create two logical variables
C
      LOGICAL LOG1,LOG2
      DATA ILOG1,ILOG2/.TRUE.,.FALSE./
C
C*   Initialize plotter
C
      CALL PLTON(IBUF,512,3)
C
C*   Create a border by filling the screen then erasing
the drawing zone
C
      CALL FILSEL(ILOG1)
      CALL XARECT(0,0,1023,779)
      CALL XERART(40,40,984,740)
      CALL FILSEL(ILOG2)
C
C*   Write in a title
C
      CALL SETCHR(80,10,10)
      CALL XAMOVE(250,675)
      CALL TEXT('TEST PROGRAMME FOR  FORTRAN   GRAPHICS
PACKAGE',43)
C
C*   Draw some nested circles
C
      DO 10 I=1,5
      CALL XAMOVE(100,500)
      CALL CIRC(I*-100)
10    CONTINUE
C
C*   Draw a rectangular pattern
C
      CALL XARECT(100,100,240,200)
      CALL XAMOVE(100,100)
      CALL XRPLOT(140,100)
      CALL XRMOVE(-140,0)
      CALL XRPLOT(140,-100)
      CALL FILSEL(ILOG1)
      CALL XERART(150,130,190,170)
```

```
      C
      C*    Demonstrate the various text sizes
      C
            CALL SETCHR(5,10,10)
            CALL XAMOVE(500,550)
            CALL TEXT('THIS IS THE SMALLEST WRITING',28)
            CALL SETCHR(30,10,10)
            CALL XAMOVE(500,525)
            CALL TEXT('THIS IS THE NEXT SIZE UP',24)
            CALL SETCHR(55,10,10)
            CALL XAMOVE(500,500)
            CALL TEXT('THIS IS THE NEXT SIZE UP',24)
            CALL SETCHR(80,10,10)
            CALL XAMOVE(425,450)
            CALL SETSLN(1)
            CALL  TEXT('THIS  IS  THE  LARGEST  SIZE  WITH
      SLANT',35)
      C
      C*    Draw a chequer pattern
      C
            DO 30 J=1,5
            DO 30 I=1,4
            K=0
            IF(J.EQ.2.OR.J.EQ.4)K=50
            IX1=400+I*100+K
            IY1=50+J*50
            CALL XARECT(IX1,IY1,IX1+50,IY1+50)
      30    CONTINUE
      C
      C*    End of plot. Sign off
      C
            CALL PLTOFF
            STOP
            END
```

A screen dump of the graphical output of this program
is included as Figure 3.
Other typical examples of graphical output produced
using these routines are included as Figures 4,5,6 and 7.

## Figure 1 : Point plot markers

Type 1
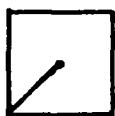
Type 2
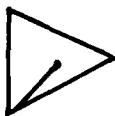
Type 3

Type 4
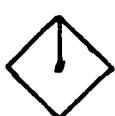
Type 5
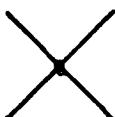
Figure 2: Schematic of parameters in AXEL
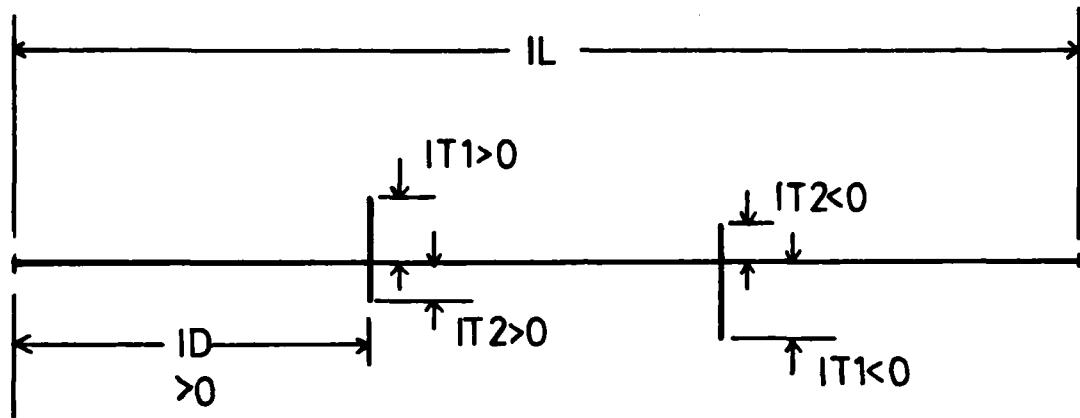
TEST PROGRAMME FOR FORTRAN GRAPHICS PACKAGE

THIS IS THE SMALLEST WRITING
THIS IS THE NEXT SIZE UP
THIS IS THE NEXT SIZE UP

*THIS IS THE LARGEST SIZE WITH SLANT*
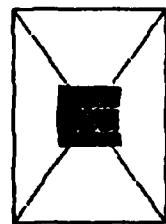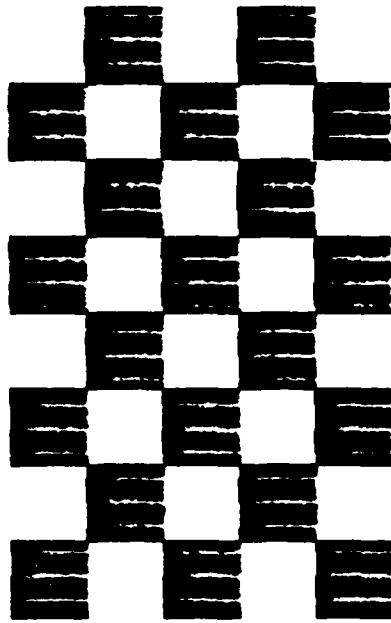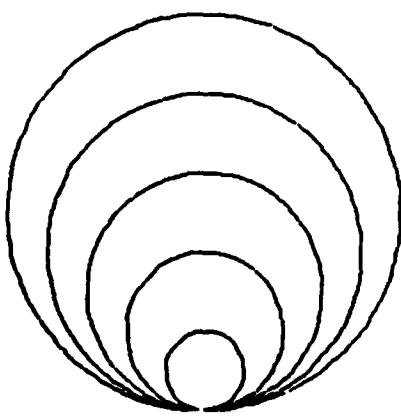
## Figure 4 : Plot of small feature cross-section
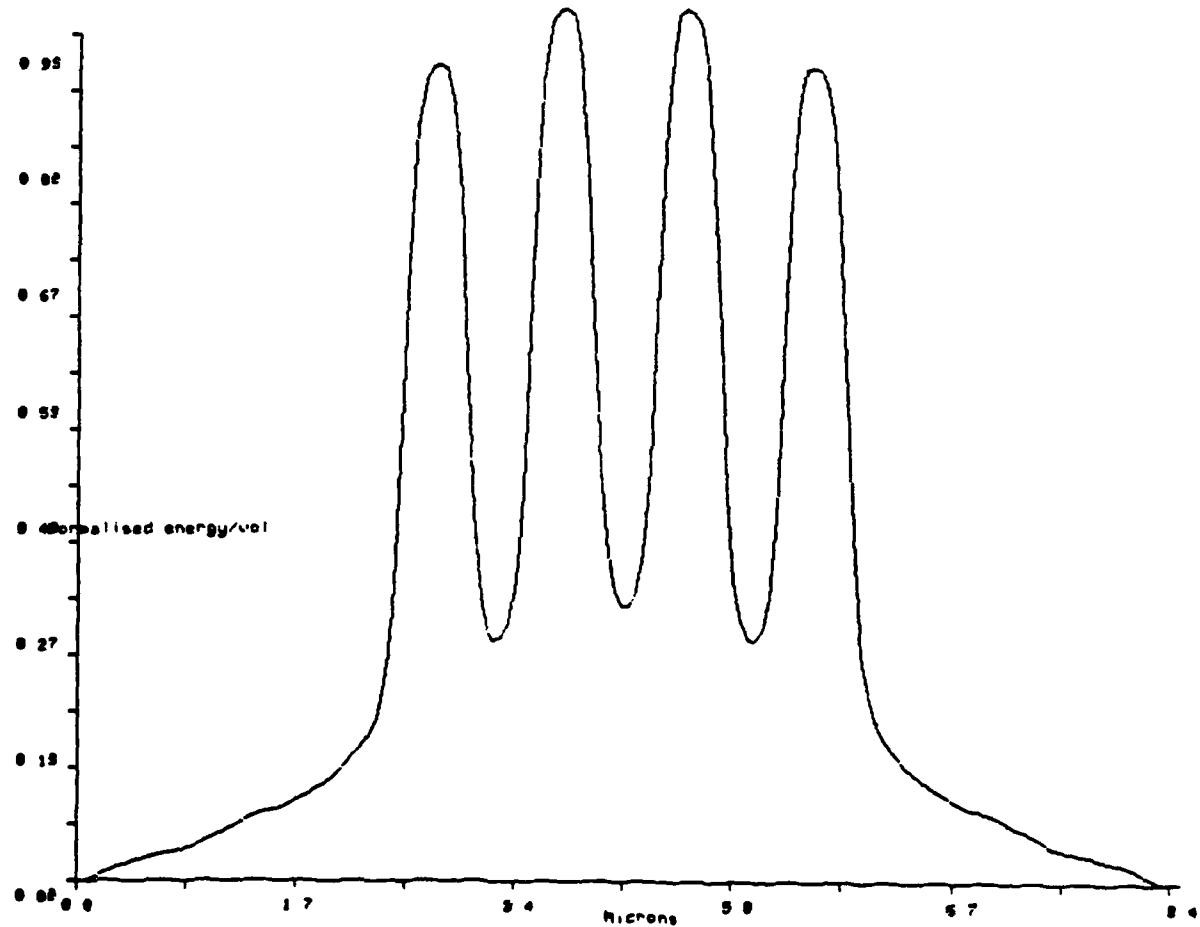
Figure 5 : Example Slice Plot

EBMF2 EXPOSURE FILE NAME =
ON 21-AUG-84
CHIP SIZE = 3 7506   2 4002
REPEAT DISTANCES = 3 7500   3500 0000

MM

0 00   10 00   20 00   30 00   40 00   50 00   60 00   70 00   80 00
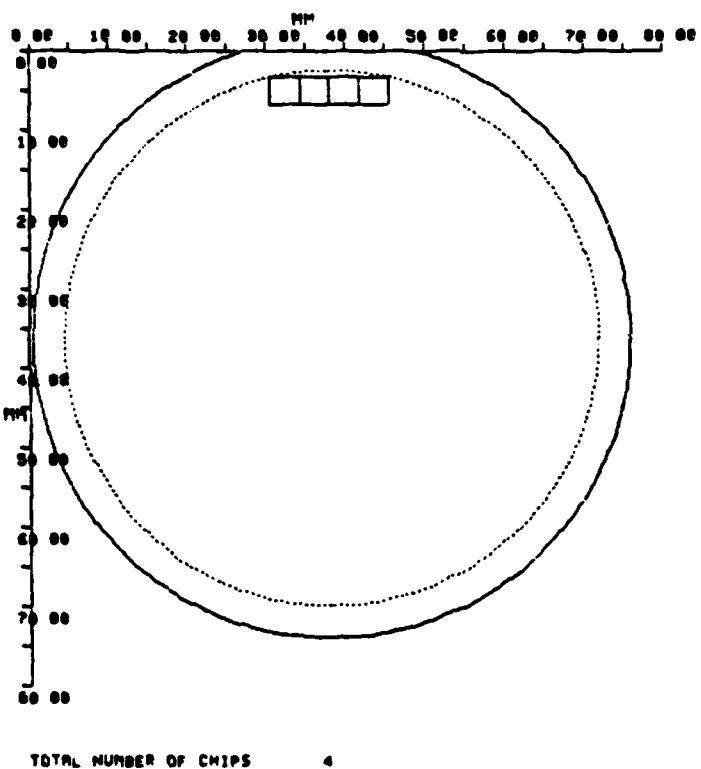
TOTAL NUMBER OF CHIPS   4

Figure 6 : Example Plot of Monte-Carlo Simulation

MONTE CARLO SIMULATION OF ELECTRON TRAJECTORIES
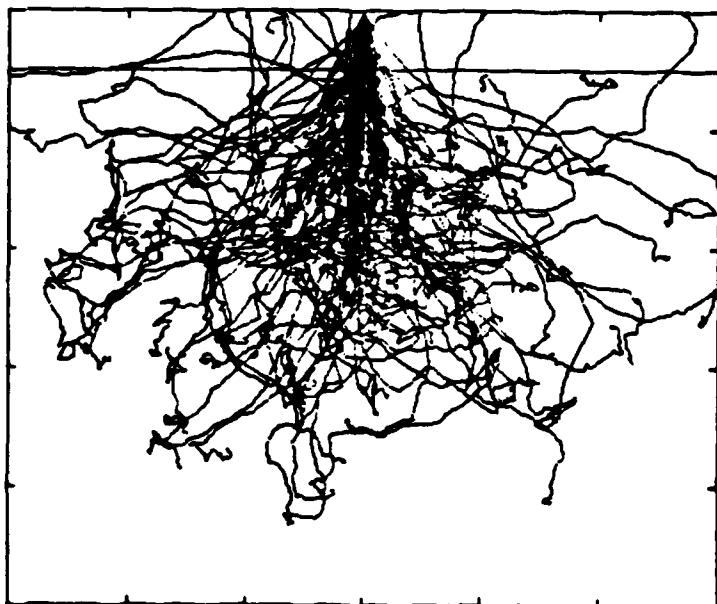
100 ELECTRONS at 20.0 keV

0.50 Microns of PMMA on SILICON

14.0 Percent BACKSCATTERED
11.0 Percent REEMERGED

1.00 Micron x grid spacings
1.00 Micron y grid spacings

Figure 7 : Example 3-D plot

DRAW3D   --   STOP END OF PLOT

ENERGY DEPOSITION SURFACE FOR 20 keU ELECTRONS
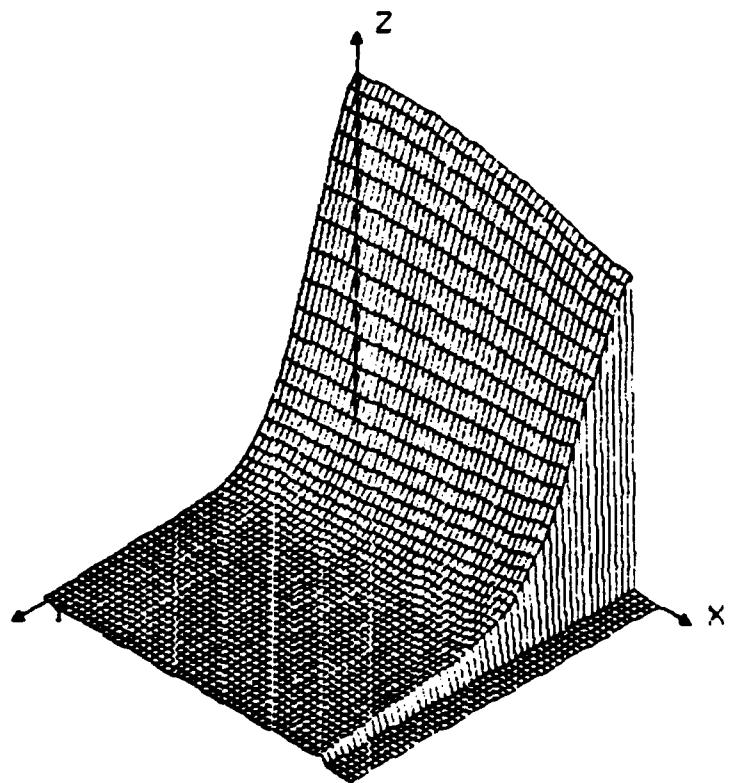
FROM   COHEN25.DAT

DOCUMENT CONTROL SHEET

Overall security classification of sheet ....

UNCLASSIFIED

(As far as possible this sheet should contain only unclassified information. If it is necessary to enter
classified information, the box concerned must be marked to indicate the classification eg (R) (C) or (S))

1. DRIC Reference (if known)	2. Originator's Reference MEMORANDUM 3739	3. Agency Reference	4. Report Security Classification Unclassified
5. Originator's Code (if known)	6. Originator (Corporate Author) Name and Location Royal Signals and Radar Establishment		
5a. Sponsoring Agency's Code (if known)	6a. Sponsoring Agency (Contract Authority) Name and Location		

7. Title

A FORTRAN PLOTTING PACKAGE FOR GRAPHIC VDU's

7a. Title in Foreign Language (in the case of translations)

7b. Presented at (for conference papers) Title, place and date of conference

8. Author 1 Surname, initials	9(a) Author 2	9(b) Authors 3,4...	10. Date	pp. ref.
V J Mifsud	C Broughton			
11. Contract Number		12. Period	13. Project	14. Other Reference

15. Distribution statement

Unlimited

Descriptors (or keywords)

continue on separate piece of paper

Abstract A set of plotting routines have been implemented in FORTRAN on
DEC LSI-11, PDP-11 and VAX-11 computers to provide a transportable graphics
capability to improve data presentation and machine control on vector scan
particle beam lithography machines. The routines are general in nature
and have already proved to be of use in many scientific applications requiring
compact and tailored graphics capabilities within specific programs. This
document is designed to act as a User's Guide for the suite. Versions of
the routines are available for ReGIS, Tektronics 4010/4014 and SIGMA native-
mode graphics terminals.

END

FILMED

1-85

DTIC